```
In [1]: import pandas as pd
         df=pd.read_csv("diabetes.csv")
 In [4]: df.shape
 Out[4]: (768, 3)
 In [3]: df.isnull().sum()
 Out[3]: Glucose
                    0
         BMI
                    0
         Outcome
                    0
         dtype: int64
 In [5]: X=df.iloc[:,:-1].to_numpy()
         y=df.iloc[:,-1].to_numpy()
 In [6]: from sklearn.model_selection import train_test_split
         X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_sta
In [12]: from sklearn.tree import DecisionTreeClassifier
         clf=DecisionTreeClassifier(random_state=0)
         clf.fit(X_train,y_train)
Out[12]:
                  DecisionTreeClassifier
         DecisionTreeClassifier(random_state=0)
In [13]:
        import matplotlib.pyplot as plt
         %matplotlib inline
         from sklearn.tree import plot tree
         plt.figure(figsize=(20,10))
         plot_tree(clf,feature_names=['Glucose','BMI'],class_names=['No','Yes'])
         plt.show()
```

```
In [15]:
                clf.set_params(max_depth=3)
Out[15]:
                                           DecisionTreeClassifier
                 DecisionTreeClassifier(max_depth=3, random_state=0)
In [16]:
                clf.fit(X_train,y_train)
                plt.figure(figsize=(20,10))
                plot_tree(clf,feature_names=['Glucose','BMI'],class_names=['No','Yes'])
                plt.show()
                                                                         Glucose <= 123.5
gini = 0.461
samples = 614
value = [393, 221]
                                                                            class = No
                                          BMI <= 26.45
gini = 0.301
samples = 352
value = [287, 65]
class = No
                                                                                                         BMI <= 30.05
gini = 0.482
samples = 262
value = [106, 156]
class = Yes
                                                         Glucose <= 99.5
gini = 0.376
samples = 255
value = [191, 64]
class = No
                                                                                        Glucose <= 145.5
                                                                                                                        Glucose <= 157.5
gini = 0.418
                          Glucose <= 106.5
gini = 0.02
                                                                                         gini = 0.438
samples = 74
value = [50, 24]
class = No
                           samples = 97
value = [96, 1]
class = No
                                                                                                                        samples = 188
value = [56, 132]
class = Yes
                                                                                  gini = 0.298
samples = 44
                     gini = 0.0
                                   gini = 0.069
                                                   gini = 0.241
                                                                   gini = 0.443
                                                                                                  gini = 0.498
                                                                                                                  gini = 0.482
                                                                                                                                  gini = 0.219
                                                  samples = 107
                    samples = 69
                                                                  samples = 148
                                                                                                  samples = 30
                                                                                                                 samples = 116
                                   samples = 28
                                                                                                                                 samples = 72
                   value = [69, 0]
                                   value = [27, 1]
                                                  value = [92, 15]
                                                                  valuė = [99, 49]
                                                                                  value = [36, 8]
                                                                                                  value = [14, 16]
class = Yes
                                                                                                                 value = [47, 69]
class = Yes
                                                                                                                                 value = [9, 63]
class = Yes
                    class = No
                                                    class = No
                                                                    class = No
                predictions=clf.predict(X_test)
In [17]:
               clf.predict([[90,20],[200,30]])
In [18]:
Out[18]: array([0, 1], dtype=int64)
In [19]:
                from sklearn.model_selection import cross_val_score
                scores=cross_val_score(clf,X_train,y_train,cv=5,scoring='accuracy')
                accuracy=scores.mean()
                accuracy
Out[19]: 0.7182993469278955
In [20]:
               from sklearn import metrics
                cf=metrics.confusion_matrix(y_test,predictions)
                cf
Out[20]: array([[90, 17],
                            [20, 27]], dtype=int64)
```

```
In [22]:
         tp=cf[1][1]
         tn=cf[0][0]
         fp=cf[0][1]
         fn=cf[1][0]
         print(f"tp:{tp}, tn:{tn},fp:{fp},fn:{fn}")
         tp:27, tn:90,fp:17,fn:20
In [24]: print("accuracy", metrics.accuracy_score(y_test, predictions))
         accuracy 0.7597402597402597
In [25]: print("Precision", metrics.precision_score(y_test, predictions))
         Precision 0.6136363636363636
In [27]: print("Recall", metrics.recall_score(y_test, predictions))
         Recall 0.574468085106383
         feature_importances = clf.feature_importances_
In [29]:
         print("Feature importances:",feature_importances)
         Feature importances: [0.74096359 0.25903641]
 In [ ]:
```